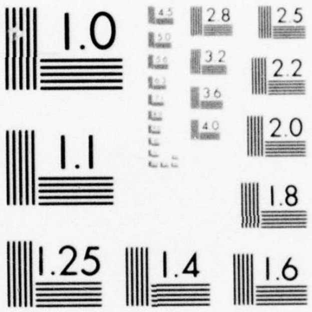
AD-A049 710   RENSSELAER POLYTECHNIC INST TROY N Y COMPUTER RESEAR--ETC F/G 9/2
PROGRAMMER'S MANUAL FOR THE CHAP LINE-DRAWING PROCESSING LANGUA--ETC(U)
JUN 77 H FREEMAN, J PAOLICELLI, M POTMESIL   AFOSR-76-2937
UNCLASSIFIED   CRL-52A   AFOSR-TR-77-0103   NL

1 OF 1
AD
A049710

END
DATE
FILMED
3-78
DDC


1.0
4.5
5.0
5.6
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

RENSSELAER POLYTECHNIC INSTITUTE
KNOWLEDGE AND THOROUGHNESS
1824

Rensselaer Polytechnic Institute

Troy, New York 12181

2

Technical Report CRL-52A

PROGRAMMER'S MANUAL FOR THE CHAP LINE-DRAWING PROCESSING LANGUAGE (Revised)

by

H. Freeman
J. Paolicelli
M. Potmesil

June 1977

Prepared for
Directorate of Mathematical and Information Sciences
Air Force Office of Scientific Research
Air Force Systems Command, USAF
Grant Number AFOSR 76-2937

Rensselaer Polytechnic Institute

TROY, NEW YORK 12181

Supersedes
A036391

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER 19 — 18 AFOSR-TR-77-0103

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER — 14 CRL-52A

4. TITLE (and Subtitle) — 6 PROGRAMMER'S MANUAL FOR THE CHAP LINE-DRAWING PROCESSING LANGUAGE (REVISED).

5. TYPE OF REPORT & PERIOD COVERED — 9 Technical rept., ~~Interim~~

6. PERFORMING ORG. REPORT NUMBER — CRL-52A

7. AUTHOR(s) — 10 H./Freeman, J./Paolicelli, M./Potmesil

8. CONTRACT OR GRANT NUMBER(s) — 15 ✓ AFOSR-76-2937, ✓ AFOSR-75-2755

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Rensselaer Polytechnic Institute
Electrical & Systems Engineering Department
Troy, NY 12181
CRL

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS — 61102F 16 2304 17 A2

11. CONTROLLING OFFICE NAME AND ADDRESS
Air Force Office of Scientific Rsch/NM
Bolling AFB, DC 20332

12. REPORT DATE — 11 June 1977

13. NUMBER OF PAGES — ~~82~~ 12 88p.

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)

15. SECURITY CLASS. (of this report) — UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Supersedes A036391
mc

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes CHAP, a computer language for processing line drawings represented in terms of the 8-direction chain code. The language consists of a set of FORTRAN subroutines, most of which are machine independent. The subroutines may be called by a user program to accomplish specific line-drawing processing tasks. The chain data is stored in packed (machine-dependent) form and is unpacked for processing. The report lists all currently available CHAP subroutines and provides all necessary information for a CHAP programmer.

409952

UNCLASSIFIED mc next page

20. Abstract

Specific information for different machine implementations are given in the appendices. Implementations exist for IBM 360 , UNIVAC 1108/1110, CDC 6600, and ADAGE AGT 30/130. Versions for other machines can be readily generated.

ABSTRACT

This report describes CHAP, a computer language for processing line drawings represented in terms of the 8-direction chain code. The language consists of a set of FORTRAN subroutines, most of which are machine independent. The subroutines may be called by a user program to accomplish specific line-drawing processing tasks. The chain data is stored in packed (machine-dependent) form and is unpacked for processing. The report lists all currently available CHAP subroutines and provides all necessary information for a CHAP programmer. Specific information for different machine implementations are given in the appendices. Thus far an implementation exists only for an IBM 360 computer. However, implementations for UNIVAC 1108/1110, CDC 6600, and ADAGE AGT 30/130 are presently being planned.

ACKNOWLEDGMENT

The authors wish to acknowledge the contributions of many of their co-workers, assistants and graduate students at both Rensselaer Polytechnic Institute and New York University, who have assisted in the preparation of CHAP. Special thanks are due to Kathleen Clare, Albert Eng, Andrew Fogel, Gerald J. Libby, Christine Mancuso, Robert Marshall, and Andrew Robinson.

The work was supported by the Directorate of Mathematical and Information Sciences, Air Force Office of Scientific Research, AFSC, under grants AFOSR 75-2755 and AFOSR 76-2937. This support is gratefully acknowledged.

TABLE OF CONTENTS

1. INTRODUCTION

Since the introduction of the chain code representation for free-form line drawings in 1961 [1], many algorithms for processing chain-encoded line drawings have been developed [2-4]. CHAP is a result of an effort to consolidate these algorithms into a single line-drawing processing language. The language consists of a set of FORTRAN subroutines which can be invoked by users' programs. The design of CHAP is analogous to that of the PAX image processing language [5]. The CHAP subroutines are mainly written in standard ANSI FORTRAN IV and implemented on an IBM 360/67 computer. A study was made to determine how CHAP could be made readily portable to other computers. At issue here were questions concerning differences in word length (ranging from 16 to 60 bits), subtle difference in the particular FORTRAN versions, variations in I/O handling, and problems of linking to other image processing software.

2. THE CHAP LANGUAGE

A chain is a sequence of octal digits a_i, $i = 1,\ldots.,n$, representing a curve in terms of a string of short, connected, straight-line segments. Each octal digit a_i corresponds to a particular directed line segment (chain link) of length $T(\sqrt{2})^s$ oriented at an angle of $m(45^o)$ relative to the positive x axis; m is the octal value of a_i, s is the modulo-two value of m, and T is a scale factor. A chain may be regarded as a sequence of line segments connecting nodes that are 8-adjacent to each other on a square lattice of spacing T. Even-valued chain links connect nodes lying vertically or horizontally adjacent and are of length T; odd-valued links connect nodes lying diagonally adjacent and are of length $T\sqrt{2}$. The scale factor T is normally set to unity. A chain usually contains special signal codes (i.e., control codes) distinguished by the link sequence $04L_1L_2$. Page 4.1 lists the presently assigned chain signal codes.*

In CHAP, chain links are stored in two formats--packed and unpacked. In the packed format, the chain links are stored in an array with $\lfloor n/3 \rfloor$ links in one word, where n is the number of bits per word. Thus for the CDC-6600 implementation, twenty links would be stored in one 60-bit word. A chain begins with a chain identification number (which may be up to 63 octal digits long) followed by the x and y coordinates of its initium. For example, the sequence '04070510000042600035042700025.....' indicates that chain number 1000_8 follows, and that the coordinates of its initium are $x=35_8$ and $y=25_8$. A chain is terminated by the end-of-chain indicator '0400'.

* A reader unfamiliar with the chain coding scheme is advised to read reference 4, which provides a good introductory survey of the subject.

CHAP maintains a catalog of all chains currently in memory. The catalog entry for a chain contains the chain identification number and the pointers to the first and last memory locations of the chain. This arrangement allows a user to specify which chains are to be processed by the various routines by simply referring to the chain identification numbers. When a chain is to be processed by a routine, it is unpacked from the packed format into a working buffer. Usually only a part of the chain in unpacked at a time, processed, and then the next part of the chain is unpacked. There are three working buffers in CHAP. Their contents are also catalogued by CHAP. This avoids unpacking a chain which has already been unpacked into a working buffer by a previous call to a processing routine. Since some routines, such as chain crosscorrelation and chain intersection, operate on two chains simultaneously, they use two of the working buffers at the same time. The unpacking routine of CHAP stores one link per word, except that the signal codes are stored with the two-digit code L_1L_2 biased by 10_{10} when the signal code contains no data, and biased by 100_{10} when it does, in which case the data is stored in the next memory location(s). Thus the chain sequence given in the previous paragraph is unpacked into individual words as follows (Now shown as FORTRAN decimal integers):

107	chain-identification-number code (100_{10} +07)
4096	chain identification number ($10000_8=496_{10}$)
122	x-coordinate code ($100_{10}+26_8=122_{10}$)
29	x-coordinate ($35_8=29_{10}$)
123	y-coordinate code ($100_{10}+27_8=123_{10}$)
21	y-coordinate ($25_8=21_{10}$)

This format of unpacked working buffers allows efficient processing because values from 0 to 7 indicate chain directional links and all other values indicate signal codes; those greater than 100_{10} also have one or more following data words. Note that links preceded by a repeat signal code (such as 0417) are completely unpacked into the number of links indicated by the repeat code and, therefore, the signal code does not appear in the unpacked form.

CHAP can also be used to create a chain. First a working buffer must be initialized and a chain identification number must be assigned and entered into the buffer. This is accomplished by the CHAP subroutine BUILD. Then links or sequences of links are entered and appended to the identification number and any signal codes in the buffer. The chain may then be put into packed form and catalogued as a CHAP chain, or it may be processed by any of the CHAP chain processing routines.

3. DEFINITIONS OF CHAP TERMS

Binary Plane - A two-dimensional array of bits which are set to 1 at locations of chain vertex points and 0 everywhere else. The CHAP binary plane is stored in common block/PLANES/NX,NY,PLANE(NX*NY/NBW) where NX is the number of bits (points) in the x direction of the plane and NY is the number of bits (points) in the y direction of the plane. PLANE is a linear array in which the bits of the plane are stored. The dimension of PLANE is [NX*NY/NBW] where NBW is the number of bits per word.

Chain - A sequence of octal digits called links. A chain consists of signal codes and directional links. A chain begins with a signal code giving its identification number, usually followed by signal codes giving the initial x and y coordinates of the chain. The initial coordinates are set to zero if they are not specified. A chain must be terminated by the end-of-chain signal code '0400'.

Chain data file - A data file containing chains in the following format:

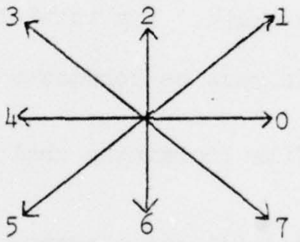
Column	
1-5	Chain identification number
6	Ignored
7-72	Chain data
73-end	Sequence numbers

A data file is terminated by a zero chain identification number. The chain data file may contain any characters but only octal digits (0 through 7) are read as chain links. This allows users to freely insert blanks or other (non-octal-digit) characters to improve the readability of chain data files.

Chain identification number - A positive integer by which a chain is uniquely identified to CHAP. In chain data files, it is a five-digit decimal number entered in columns 1-5. It may be assigned by the user when building a chain in CHAP.

Signal codes - In a chain, a sequence of links beginning with 04 that indicates the start of control information. A list of all currently assigned signal codes is given in Section 4.

Directional link - A directed straight-line segment of length 1 or $\sqrt{2}$ connecting the current node in a square lattice of unit spacing to one of its eight neighboring nodes. A link is identified according to its direction by a number from 0 to 7.



Link - An octal digit within a chain.

Packed chain - A chain whose links are packed into memory words with $\lfloor n/3 \rfloor$ links per word, where n is the number of bits per word of the particular computer being used.

Unpacked chain - A chain whose directional links are stored one per word. Signal codes that do not contain any data are biased by 10_{10} to differentiate them from directional links. Signal codes that contain data are biased by 100_{10} to differentiate them from directional links and signal codes without data. The data is stored in the locations following the signal code. For example, the chain '04070501000 042600010 042700021 1254 0400' is unpacked into individual words as follows (Now shown as FORTRAN decimal integers):

107	chain-identification-number signal code ($07+100_{10}$)
512	chain identification number ($1000_8=512_{10}$)
122	x-coordinate signal code ($26 +100_{10}=122_{10}$)
8	x-coordinate ($10_8=8_{10}$)
123	y-coordinate signal code ($27_8+100_{10}=123_{10}$)
17	y-coordinate ($21_8=17_{10}$)
1	directional link
2	directional link
5	directional link
4	directional link
10	end-of-chain signal code ($00+10_{10}$)

<u>Chain Vertex</u> - A chain vertex is a gird-node point through which the chain passes. It is normally denoted by the number of the link which departs from it. Thus vertex 1 is the initium of the chain, and the last link joins vertex n to vertex n+1. Vertex n+1 is the terminus of the chain.

4. LIST OF CHAP CHAIN SIGNAL CODES

Code	Description
0400	End-of-chain indicator.
0401	Invisible chain follows.
0402	Visible chain follows. Every chain is assumed to be visible initially.
0403	Remainder of word is to be skipped. This code is ignored by CHAP.
0404	Valid 04 sequence of directional links
0405XYZ	Point marker number $(XYZ)_8$
0407UV	Chain identification number $(UV)_8$ digits long follows.
0410WXYZ	Non-chain data $(WXYZ)_8$ digits long follows.
0411	Non-chain data follows until terminated by end-of-non-chain-data indicator.
04127777	End-of-non-chain-data indicator.
0413UWXYZ	Chain node number $(WXYZ)_8$ connects with $(U)_8$ other chains.
0414UVWXYZ	Rotation indicator.
0415UVWXYZS	Scale change indicator.
0417UXYZ	Link U is to be repeated $(XYZ)_8$ times.
0420UN	Link U is to be repeated as many times as specified by the next $(N)_8 + 4$ digits.
0421TUVWXYZ	The sequence consisting of the immediately following $(TUV)_8$ links is to be repeated $(WXYZ)_8$ times.
0422U	Color indicator.
0423WXYZ	Elevation value indicator $(WXYZ)_8$
0424XYZ	Grey-level value indicator $(XYZ)_8$
0425U	Check code. U_8 yields modulo-8 sum of zero.
0426VWXYZ	Initial x coordinate is $(VWXYZ)_8$
0427VWXYZ	Initial y coordinate is $(VWXYZ)_8$

5. LIST OF SUBROUTINES GROUPED BY FUNCTION

CHAP Initialization:

CHAPBD	Block data program. Allocate and initialize the CHAP labelled common blocks.

Subroutines used for chain synthesis:

BUILD	Start to build a chain by initializing buffer and inserting chain's identification number.
XCOORD	Append an x coordinate to a chain.
YCOORD	Append a y coordinate to a chain.
LINK	Append a link to a chain.
LINKSQ	Append a sequence of links to a chain.
INVIS	Append invisible-chain-follows signal code.
VISIBL	Append visible-chain-follows signal code.
CHLINE	Generate a sequence of links approximating a straight line.
EOCH	Append end-of-chain signal code.

Chain management subroutines:

INPUT	Input a chain data file.
CHLIST	List all chains in CHAP catalog.
DELETE	Delete a chain from CHAP.
OUTPUT	Outputs chains to I/O unit 7 in format read by INPUT.
ARRAY	Convert a chain into an array containing the links one per word.
VERTEX	Convert a chain into lists of vertex coordinates.
NPACK	Pack a chain.
UNPACK	Unpack a chain.
BPLANE	Clear binary plane and convert a chain to binary plane representation.
APLANE	Convert and append a chain into a binary plane.
BPRINT	Print the binary plane.
PRINT	Convert chain to binary plane representation and print the binary plane.
CHPAX	Convert a CHAP chain to a PAX chain.
CHPLOT	Plot a chain on a digital plotter.

Link management routines:

GTLINK	Unpack a link from a packed format chain.
OCTNUM	Form an octal number from a sequence of links.
STLINK	Store a link in a packed format buffer.

Chain analysis subroutines:

ANGLE	Compute angle between the x axis and line defined by two chain
AUTO	Compute the autocorrelation of a chain.
CENTRD	Compute the x and y coordinates of the centroid of a chain.
COARSE	Requantize chain on a coarser grid.
CROSS	Compute the crosscorrelation of two chains.
ECAREA	Compute the enclosed are of a closed chain.
INITXY	Find the x and y coordinates of the chain initium
INTERS	Find the intersection of two chains.
LENGTH	Compute the length of a chain.
LMOM1	Compute the first moment of a chain about a line.
LMOM2	Compute the second moment of a chain about a line.
LNAREA	Compute the area between a chain and a line.
LNCHD	Compute the max and min distances from a chain to a line.
MATCH	Match two chain profiles.
MAXMIN	Compute the max and min extents of a chain in x and y directions.
MOM1	Compute the first moment of a chain about 0,45,90,135 axes.
MOM2	Compute the second moment of a chain about 0,45,90,135 axes.
PDIST	Compute the distance between two points on a chain.
POLYGN	Determine a polygonal approximation of a chain to a specified precision.
PNTCHD	Compute *max and min distances from* a chain to a point.
RESID	Compute chain residue.
ROSCAL	Rotate and scale a chain.
WHEX	Compute chain width, height, 45 and 135 degrees extents.
XPROFL	Compute chain profile along x axis.
YPROFL	Compute chain profile along y axis.

6. ALPHABETICAL LIST OF GLOBAL NAMES AND COMMON BLOCKS

The following is a list of all global names defined in CHAP including Subroutines (S), Entry to a subroutine (E), Integer functions (IF), and Block data (BD).

Name	Type	Description
ANGLE	S	Compute angle between the x axis and the line defined by two chain vertices.
APLANE	E	Convert and append a chain into a binary plane.
ARRAY	S	Convert chain into array containing the links one per word.
AUTO	S	Compute the autocorrelation of a chain.
BPLANE	S	Clear binary plane and convert a chain into it.
BPRINT	E	Print binary plane.
BUILD	S	Start to build a chain by initializing buffer and inserting chain's identification number.
CENTRD	S	Compute the x and y coordinates of the centroid of a chain.
CHAPBD	BD	Allocate and initialize the CHAP labelled common blocks.
CHLINE	S	Generate a sequence of links approximating a straight line.
CHLIST	S	List all chains in CHAP catalog.
CHPAX	S	Convert a CHAP chain to a PAX chain.
CHPLOT	S	Plot a chain on a digital plotter.
COARSE	S	Requantize chain on a coarser grid.
CROSS	S	Compute the crosscorrelation of two chains.
DELETE	S	Delete a chain from CHAP.
ECAREA	S	Compute the enclosed area of a closed chain.
EOCH	S	Append end-of-chain signal code to a chain.
GTLINK	IF	Unpack a link from a packed format chain.
INITXY	S	Find the x and y coordinates of the chain initium
INPUT	S	Input a chain data file.
INTERS	S	Find the intersections of two chains.
INVIS	S	Append invisible-chain-follows signal code.
LENGTH	S	Compute the length of a chain.
LINK	S	Append a link to a chain.
LINKSQ	S	Append a sequence of links to a chain.
LMOM1	S	Compute the first moment of a chain about a line.
LMOM2	S	Compute the second moment of a chain about a line.
LNAREA	S	Compute the area between a chain and a line.
LNCHD	S	Compute the max and min distance from a chain to a line.
MATCH	S	Match two chain profiles.
MAXMIN	S	Compute the max and min extents of a chain in x and y directions.
MOM1	S	Compute the first of a chain about 0,45,90,135 axes.
MOM1A	E	Entry to routine MOM2 (CHAP internal use only).
MOM2	S	Compute the second moment of a chain about 0,45,90,135 axes.
MOM2A	E	Entry to routine MOM1 (CHAP internal use only).
NPACK	S	Pack a chain.
OCTNUM	IF	Form an octal number from a sequence of links.
OUTPUT	S	Output chains to I/O unit 7 in format read by INPUT.
PDIST	S	Compute the distance between two points on a chain.

PNTCHD	S	Compute the max and min distances from a chain to a point.
POLYGN	S	Determine a polygonal approximation of a chain to a specified precision.
PRINT	S	Convert chain to binary plane representation and print binary plane.
RESID	S	Compute chain residue.
ROSCAL	S	Rotate and scale a chain.
STLINK	S	Store a link in a packed format buffer.
UNPACK	S	Unpack a chain.
VERTEX	S	Convert a chain into lists of vertex coordinates.
VISIBL	S	Append visible-chain-follows signal code.
WHEX	S	Compute chain width, height, 45 and 135 degree extents.
XCOORD	S	Append an x coordinate to a chain.
XPROFL	S	Compute chain profile along x axis.
YCOORD	S	Append a y coordinate to a chain.
YPROFL	E	Compute chain profile along y axis.

The following labelled common blocks are defined in CHAP:

CHAPMC
UCHAIN
PLANES
XYCOMP
ICHAIN
IOBUFF
PCHAIN
WCHAIN
STATUS

Caution: Note that users must not define these global names and common block names in their programs.

7. DESCRIPTIONS OF CHAP SUBPROGRAMS

The following is an alphabetically ordered description of all CHAP subprograms. The FORTRAN data types of parameters passed to CHAP subroutines and functions are abbreviated in column Type as follows:

I	Integer
R	Real
L	Logical
H	Hollerith character string

The parameters passed to CHAP subroutines are also classified in column headed I/O as follows:

I Input--value is assigned to this parameter in the user's program. It is not changed by CHAP programs.

O Output--value is assigned to this parameter in a CHAP program.

T Temporary--value is undetermined before and after calls to CHAP programs. The parameter is used for temporary storage by CHAP.

In the column headed Dimension, a "-" indicates the parameter is not an array but is a one-word variable.

NAME: ANGLE

TITLE: Compute angle between the positive x axis and line defined by two chain vertices.

TYPE: Subroutine.

CALL FORM: CALL ANGLE(IDENT,NP1,NP2,A)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
NP1	I	I	-	First chain vertex
NP2	I	I	-	Second chain vertex
A	O	R	-	Computed angle (in degrees)

COMMON BLOCKS: XYCOMP, UCHAIN

SUBROUTINES CALLED: UNPACK

DESCRIPTION: The angle (in degrees) between the positive x-axis and the straight line defined by the two chain vertices NP1 and NP2 is computed.

WORKING BUFFER USED: 1

NAME: APLANE

TITLE: Convert and append a chain into a binary plane.

TYPE: Entry to subroutine BPLANE.

CALL FORM: CALL APLANE(IDENT)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number

COMMON BLOCKS: CHAPMC, XYCOMP, UCHAIN, PLANES

SUBROUTINES CALLED: Implementation-dependent. See appropriate appendix.
UNPACK

DESCRIPTION: This subroutine converts a chain into a binary plane and stores it into array PLANE in the commom block /PLANES/. The previous contents of PLANE are not affected.

WORKING BUFFER USED: 1

NAME: ARRAY

TITLE: Convert chain into array containing the links one per word.

TYPE: Subroutine.

CALL FORM: CALL ARRAY(IDENT,LIST,N,L)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
LIST	I	I	N	An array for chain links
N	I	I	-	Dimension of LIST
L	O	I	-	Number of chain links stored in LIST

COMMON BLOCKS: UCHAIN

SUBROUTINES CALLED: UNPACK

DESCRIPTION: This subroutine stores the links of a chain into array LIST. One link, valued 0 to 7, is stored into one word of the array. All signal codes are skipped. If there are more than N links in the chain, the remaining links are skipped and a warning message is printed. L is set to number of links stored in LIST.

WORKING BUFFER USED: 3

NAME: AUTO

TITLE: Compute the autocorrelation of a chain.

TYPE: Subroutine.

CALL FORM: CALL AUTO(IDENT,ACORR,PERIOD,LIST,N)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I		Chain identification number
ACORR	O	R		Autocorrelation value
PERIOD	O	I		Period of maximum autocorrelation
LIST	T	I	N	Array to store chain directional links
N	I	I		Dimension of LIST

COMMON BLOCKS: none

SUBROUTINES CALLED: ARRAY

DESCRIPTION: This subroutine computes the maximum autocorrelation value of a chain. The autocorrelation value ACORR is in the range < 0.0,1.0>. PERIOD is the number of links of the period of the autocorrelation function when ACORR is maximum.

WORKING BUFFER USED: 3

LIMITATIONS: Note that the array LIST must be large enough to store all directional links of the chain in one link-per-word format.

NAME: BPLANE

TITLE: Clear binary plane and convert a chain into it.

TYPE: Subroutine.

CALL FORM: CALL BPLANE(IDENT)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number

COMMON BLOCKS: CHAPMC, XYCOMP, UCHAIN, PLANES

SUBROUTINES CALLED: Implementation-dependent. See appropriate appendix.
UNPACK

DESCRIPTION: This subroutine clears the contents of array PLANE in the common block /PLANES/ and then converts chain IDENT into a binary plane representation and stores it in PLANE.

WORKING BUFFER USED: 1

NAME: BPRINT

TITLE: Print binary plane.

TYPE: Subroutine entry point in PRINT.

CALL FORM: CALL BPRINT(IW)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IW	I	I	4	Specified window to be printed

COMMON BLOCKS: CHAPMC, IOBUFF, PLANES

SUBROUTINES CALLED: Implementation-dependent. See appropriate appendix.

BPLANE

DESCRIPTION: A window specified by IW of the CHAP's binary plane PLANE is common block /PLANES/ is printed.

The contents of IW are

IW(1) - initial x point
IW(2) - final x point
IW(3) - initial y point
IW(4) - final y point

NAME: BUILD

TITLE: Start to build a chain by initializing buffer and inserting chain's identification number.

TYPE: Subroutine.

CALL FORM: CALL BUILD(IBUFF,IDENT)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IBUFF	I	I	-	CHAP buffer to store the chain
IDENT	I	I	-	Chain identification number

COMMON BLOCKS: STATUS, UCHAIN,WCHAIN

SUBROUTINES CALLED: none

DESCRIPTION: This subroutine initializes one of the three working buffers IBUFF in CHAP, and starts to build chain number IDENT by inserting the chain's identification number into the buffer. Also, an end-of-chain code is placed in the first position of the buffer.

NAME: CENTRD

TITLE: Compute the x and y coordinates of the centroid of a chain.

TYPE: Subroutine.

CALL FORM: CALL CENTRD(IDENT,CX,CY)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
CX	O	R	-	Centroid x coordinate
CY	O	R	-	Centroid y coordinate

COMMON BLOCKS: none

SUBROUTINES CALLED: ECAREA, MOM1

LIMITATIONS: Chain IDENT must be a closed chain.

DESCRIPTION: The x and y coordinates (CX,CY) of the centroid of a chain are computed as the ratios of the chain's moments about the x and y axes to the enclosed area.

WORKING BUFFER USED: 1

NAME: CHAPBD

TITLE: Allocate and initialize the CHAP labelled common blocks.

TYPE: Block data program.

CALL FORM: Implementation dependent. See appropriate appendix.

DESCRIPTION: This program allocates and initializes the CHAP labelled common blocks. These blocks contain machine-dependent constants, chain buffers, an input/output buffer, binary plane buffer, catalogs of chains in memory and parameters referring to the sizes of the above buffers. Changing these parameters allows a user to alter the sizes of buffers in the entire CHAP system.

The following are common blocks allocated and initialized by this program:

a) Machine dependent constants - block CHAPMC

```
/CHAPMC/ NBW,NLW,BLANK,EX,DIGITS(8)
```

NBW	number of bits per machine word
NLW	number of octal links per machine word
BLANK	left justified code of the space character
EX	left justified code of the character used to print chains from binary plane
DIGITS(8)	left justified codes of the eight octal digits

b) Components of chain directional links - block XYCOMP

```
/XYCOMP/ AX(8),AY(8)
```

AX(8)	x components of chain links 0 to 7
AY(8)	y components of chain links 0 to 7

c) Catalog of chains in CHAP - block ICHAIN

/ICHAIN/ UCH(9), MAXNCH,NCH,CHDIR(3*MAXNCH)

UCH(9) not used, replaced by /STATUS/

MAXNCH maximum allowable number of chains in CHAP memory

NCH current number of chains in CHAP memory

CHDIR(3*MAXNCH) catalog of packed chains in memory:

CHDIR(3i-2) identification number of the i-th chain in /PCHAIN/

CHDIR(3i-1) index of first location of the i-th chain in /PCHAIN/

CHDIR(3i) index of last location of the i-th chain in /PCHAIN/

d) Packed chains and working buffers - blocks PCHAIN and UCHAIN

/PCHAIN/ BUFFER(PSIZE)

BUFFER(PSIZE) array for storage of packed chains

/UCHAIN/ UNBUFF(USIZE)

UNBUFF(USIZE) array for storage of working buffers

Both arrays are to be dimensioned by the user according to his requirements.

e) Starting indices of the three working buffers - block WCHAIN

/WCHAIN/ WBASE(4)

WBASE(4) starting indices of the three partitioned working buffers in /UCHAIN/ UNBUFF(USIZE). WBASE(4) contains the dimension of UNBUFF plus one.

f) Binary plane - block PLANES

/PLANES/ NX,NY,PLANE(NX/NBW*NY)

NX	number of bits (points) in the x direction
NY	number of bits (points) in the y direction
PLANE	binary plane

g) Input/output buffer - block IOBUFF

/IOBUFF/ IO(IOSIZE)

IO(IOSIZE)	input/output buffer - usually dimensioned to 132

h) Status of unpacked chains - block STATUS

/STATUS/ ID(3),SX(3),SY(3),SSTART(3),SSTOP(3),LLINK(3),
PART(3),SEOCH(3)

The description of the following arrays applies to the chain currently contained in each working buffer.

ID(3)	the chain identification number
SX(3)	the initial x coordinate
SY(3)	the initial y coordinate
SSTART(3)	the first position used in the working buffer
SSTOP(3)	the last position used in the working buffer
LLINK(3)	the total number of links in the chain
PART(3)	the number of calls to the routine UNPACK in the current unpacking sequence (in the case of a chain which is larger than the working buffer)
SEOCH(3)	a logical flag which indicates if the end-of-chain indicator was found

REFERENCES: See appropriate appendix for a listing of this program.

NAME: CHLINE

TITLE: Generate a sequence of links approximating a straight line.

TYPE: Subroutine.

CALL FORM: CALL CHLINE(DELX,DELY,I)

PARAMETERS:

Name	I/O	Type	Dimension	Description
DELX	I	I		Final x coordinate
DELY	I	I		Final y coordinate
I	I	I		Specified starting location in the array UNBUFF to store the link sequence.

COMMON BLOCKS: /UCHAIN/UNBUFF(USIZE)

SUBROUTINES CALLED: none

DESCRIPTION: This subroutine generates a sequence of chain links, which approximate a straight line from grid point(0,0) to the point(DELX,DELY). The sequence of links is stored in array UNBUFF of common block /UCHAIN/; locations I through I + MAX(|DEXL|, |DELY|)-1. Note that end-of-chain code is not appended.

REFERENCES: Bresenham, J. E., "Algorithm for computer control of a digital plotter", IBM Systems J., (4), 1965, pp. 25-30.

NAME: CHLIST

TITLE: List all chains in CHAP catalog.

TYPE: Subroutine.

CALL FORM: CALL CHLIST

PARAMETERS: none

COMMON BLOCKS: CHAPMC,PCHAIN,ICHAIN,IOBUFF

SUBROUTINES CALLED: GTLINK

DESCRIPTION: This subroutine lists on FORTRAN logical I/O unit 6 the contents of the catalog of chains and the contents of all chains in memory at the time of the listing.

NAME: CHPAX

TITLE: Convert a CHAP chain to a PAX chain.

TYPE: Subroutine.

CALL FORM: CALL CHPAX(IDENT,LIST,N)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
LIST	O	I	N+1	PAX chain array
N	O	I	-	Number of chain direction codes in LIST

COMMON BLOCKS: UCHAIN

SUBROUTINES CALLED: UNPACK

DESCRIPTION: This subroutine converts a CHAP chain into a PAX formatted list of directional codes. Such a list can be stored into a PAX plane by the PAX routine CHAIN. The PAX direction codes are as follows:

```
6 7 8
5 * 1
4 3 2
```

The codes are stored one per word into array LIST and terminated by a zero code. N+1 is the number of codes in LIST including the terminating zero code.

WORKING BUFFER USED: 1

NAME: CHPLOT

TITLE: Plot a chain on a digital plotter.

TYPE: Subroutine.

CALL FORM: CALL CHPLOT(IDENT,STRING,SIZE)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I		Chain file
STRING	I	H	40 char	Output identification string
SIZE	I	R		Grid size in inches

COMMON BLOCKS: XYCOMP,UCHAIN

SUBROUTINES CALLED: Implementation dependent. See appropriate appendix.
UNPACK

DESCRIPTION: This subroutine plots a chain number IDENT on a digital plotter. The plotter output is identified by a 40 character Hollerith string presumably containing the user's name and account number.

WORKING BUFFER USED: 1

NAME: CHPLOT

TITLE: Plot a chain on a digital plotter.

TYPE: Subroutine.

CALL FORM: CALL CHPLOT (IDENT, GSIZE)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
GSIZE	I	R	-	Grid size in inches

COMMON BLOCKS: XYCOMP, UCHAIN

SUBROUTINES CALLED: UNPACK

WORKING BUFFER USED: 1

DESCRIPTION: This subroutine plots a chain on a digital plotter. The chain identification number is printed at the start of the chain. The calcomp routines PLOTS, PLOT, SYMBOL, and NUMBER are used.

NAME: COARSE

TITLE: Requantize chain on a coarser grid.

TYPE: Subroutine.

CALL FORM: CALL COARSE (J,XCOOR,YCOOR,N,LINKS

PARAMETERS:

Name	I/O	Type	Dimension	Description
J	I	I	-	Number of links in chain
XCOOR	I	I	J+1	X coordinates of chain vertices
YCOOR	I	I	J+1	Y coordinates of chain vertices
N	I	I	-	Number of times to expand chain grid
LINKS	O	I	J	Links of compressed chain
M	O	I	-	Number of links in LINKS

COMMON BLOCKS: none

SUBROUTINES CALLED: none

DESCRIPTION: This subroutine compresses a chain by superimposing it on a chain grid expanded N times and then requantizing it. The directional links of the new chain are stored one link per word in array LINKS.

NAME: CROSS

TITLE: Compute the crosscorrelation of two chains.

TYPE: Subroutine.

CALL FORM: CALL CROSS(IDENT1,IDENT2,CORR,J,LIST,N)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT1	I	I	-	Chain 1 identification number
IDENT2	I	I	-	Chain 2 identification number
CORR	O	R	-	Correlation between chain 1 and 2
J	I	I	-	Chain shift value
LIST	T	I	N	Array to store links of chains 1 and 2
N	I	I	-	Dimension of LIST

COMMON BLOCKS: none

SUBROUTINES CALLED: ARRAY

DESCRIPTION: This subroutine determines the degree of shape similarity (crosscorrelation) between two chains by comparing the directional links of chain 1 and chain 2 in pairs. The increment value J determines the degree of shift to the right of chain 1 with respect to chain 2. The result is the average pairwise difference between chains 1 and 2 for a given value of J. The value of CORR is in the range -1.0,1.0 .

LIMITATIONS: Note that the array LIST must be large enough to store all directional links of both chains in one-link-per-word format.

WORKING BUFFER USED: 3

6/77

NAME: DELETE

TITLE: Delete a chain from CHAP.

TYPE: Subroutine.

CALL FORM: CALL DELETE(IDENT)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number

COMMON BLOCKS: ICHAIN, PCHAIN

SUBROUTINES CALLED: none

DESCRIPTION: This subroutine deletes the chain number IDENT from the catalog of chains in CHAP.

NAME: ECAREA

TITLE: Compute the enclosed area of a closed chain.

TYPE: Subroutine.

CALL FORM: CALL ECAREA(IDENT,S)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
S	O	R	-	Enclosed area

COMMON BLOCKS: XYCOMP, UCHAIN

SUBROUTINES CALLED: UNPACK

LIMITATIONS: Chain IDENT must be closed.

DESCRIPTION: This subroutine computes the enclosed area of a chain. It is assumed that the chain is closed. The area is positive if it is enclosed in the clockwise sense and negative if it is enclosed in the contraclockwise sense.

WORKING BUFFER USED: 1

NAME: EOCH

TITLE: Append end-of-chain signal code to a chain.

TYPE: Subroutine.

CALL FORM: CALL EOCH(IDENT)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number

COMMON BLOCKS: STATUS, UCHAIN

SUBROUTINES CALLED: none

DESCRIPTION: The end-of-chain control code '0400' is appended to chain IDENT which is currently being built by CHAP, by placing the number 10_{10} in the next position of the working buffer.

NAME: GTLINK

TITLE: Unpack a link from a packed format chain.

TYPE: Integer Function.

CALL FORM: Variable=GTLINK(BUFFER,I)

PARAMETERS:

Name	I/O	Type	Dimension	Description
BUFFER	I	I	M	One dimensional array
I	I	I	-	Sequential position of the digit in array BUFFER

COMMON BLOCKS: CHAPMC

SUBROUTINES CALLED: Implementation dependent. See appropriate appendix.

DESCRIPTION: This integer function returns the I-th digit from the one dimensional array containing links in packed format, with NLW octal digits per word.

LIMITATIONS: In order to stay within the array BUFFER:

$$1 \leq I \leq NLW*M$$

NAME: INITXY

TITLE: Find the x and y coordinates of the chain initium.

TYPE: Subroutine.

CALL FORM: CALL INITXY(IDENT,X,Y)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
X	O	I	-	The initial x coordinate
Y	O	I	-	The initial y coordinate

COMMON BLOCKS: STATUS

SUBROUTINES CALLED: UNPACK

DESCRIPTION: This subroutine searches a chain for the signal codes that give the x and y coordinates of the initium. Their values are returned as X and Y. If none are found, the values of X and Y are set to zero. If there is more than one x or y coordinate in a chain, the value of the first is returned and all others are ignored.

WORKING BUFFER USED: 1

NAME: INPUT

TITLE: Input a chain data file.

TYPE: Subroutine.

CALL FORM: CALL INPUT(ICODE)

PARAMETERS:

Name	I/O	Type	Dimension	Description
ICODE	I	I	-	Catalog flag

COMMON BLOCKS: CHAPMC,PCHAIN,ICHAIN,IOBUFF

SUBROUTINES CALLED: STLINK

LIMITATIONS: Input chains must be in the form specified below.

DESCRIPTION: This subroutine reads one or more chains from a data file which has been assigned the FORTRAN logical I/O unit number 5. If the parameter ICODE is zero the catalog of chains in CHAP is initialized, else the chains read are added to those chains already catalogued by CHAP.

The chain data files must be entered in the following format:

Columns	1-5	Chain identification number
	6	Ignored
	7-72	Chain data
	73-80	Sequence number

In the chain data field all characters other than octal digits 0 to 7 are ignored. This allows users to freely insert blanks or other characters in order to improve the readability of chains. The data file is terminated by a zero chain identification number.

NAME: INTERS

TITLE: Find the intersections of two chains.

TYPE: Subroutine.

CALL FORM: CALL INTERS(IDENT1,IDENT2,ITYPE,INLCHA,INLCHB,NUMINT,X,Y,LEXIST,D)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT1	I	I	-	Chain 1 identification number
IDENT2	I	I	-	Chain 2 identification number
ITYPE	O	I	J	Type of intersection
INLCHA	O	I	J	Link of chain 1 in intersection
INLCHB	O	I	J	Link of chain 2 in intersection
NUMINT	O	I	-	Number of intersections
X	T	I	D	Array for x coordinates of chains 1 and 2
Y	T	I	D	Array for y coordinates of chains 1 and 2
LEXIST	O	I	J	Array for pointers to intersections
D	I	I	-	Dimensions of X, Y

COMMON BLOCKS: none

SUBROUTINES CALLED: VERTEX

LIMITATIONS: J must be at least the number of intersections. The dimension D of arrays X and Y must be large enough to store the coordinates of all the links of both chains, which is the number of links in both chains plus two.

DESCRIPTION: This subroutine finds all intersections between chains 1 and 2. An intersection may be of two types - nodal ITYPE=1 and non-nodal ITYPE=2. The number of intersections found is NUMINT. X,Y and LEXIST are arrays used by this subroutine to store the x and y coordinates of vertices of the two chains and pointers to intersections. INLCHA(J) and INLCHB(J) contain the number of directional links of chain 1 and 2 involved in the j-th intersection found.

WORKING BUFFER USED: 1

NAME: INVIS

TITLE: Append invisible-chain-follows signal code.

TYPE: Subroutine.

CALL FORM: CALL INVIS(IDENT)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number

COMMON BLOCKS: STATUS, UCHAIN

SUBROUTINES CALLED: none

DESCRIPTION: The invisible-chain-follows control code '0401' is appended to chain IDENT which is currently being built by CHAP, by placing the number 11_{10} in the next position of the working buffer.

NAME: LENGTH

TITLE: Compute the length of a chain.

TYPE: Subroutine.

CALL FORM: CALL LENGTH(IDENT,CHL,LCF)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
CHL	O	R	-	Computed length of a chain
LCF	I	I	-	LENGTH computation flag

COMMON BLOCKS: UCHAIN

SUBROUTINES CALLED: UNPACK

DESCRIPTION: This subroutine computes the length of a chain. All even valued links are of length 1 unit; all odd valued links are of length $\sqrt{2}$ units.If the length of both visible and invisible links is desired LCF should be set to 0. For the length of only visible links LCF should equal 1, and for only invisible links LCF should equal 2.

WORKING BUFFER USED: 1

NAME: LINK

TITLE: Append a link to a chain.

TYPE: Subroutine.

CALL FORM: CALL LINK(IDENT,LINK,NA)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
LINK	I	I	-	Link to be appended
NA	I	I	-	Number of times the link is to be appended

COMMON BLOCKS: STATUS, UCHAIN

SUBROUTINES CALLED: none

DESCRIPTION: This subroutine appends the directional link LINK to chain IDENT currently being built by CHAP. The link LINK is appended NA times, where NA > 0.

TITLE: Append a sequence of links to a chain.

TYPE: Subroutine.

CALL FORM: CALL LINKSQ(IDENT,LIST,NL,NA)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
LIST	I	I	NL	An array of links to be appended
NL	I	I	-	Number of links in LIST
NA	I	I	-	Number of times the sequence of links in LIST is to be appended

COMMON BLOCKS: STATUS,UCHAIN

SUBROUTINES CALLED: none

DESCRIPTION: This subroutine appends the sequence of NL directional links stored in array LIST, one link per word, to chain IDENT currently being built by CHAP. The sequence of links is appended NA times, where NA > 0.

NAME: LMOM1

TITLE: Compute first moment of a chain about a line.

TYPE: Subroutine.

CALL FORM: CALL LMOM1(IDENT,X1,Y1,X2,Y2,FMNT)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
X1	I	R	-	X coordinate of first point of line
Y1	I	R	-	Y coordinate of first point of line
X2	I	R	-	X coordinate of second point of line
Y2	I	R	-	Y coordinate of second point of line
FMNT	O	R	-	Computed first moment

COMMON BLOCKS: none

SUBROUTINES CALLED: LNAREA, CENTRD

LIMITATIONS: Chain IDENT must be a closed chain.

DESCRIPTION: This subroutine computes the first moment of a chain about a line defined by points (X1,Y1) and (X2,Y2).

WORKING BUFFER USED: 1

NAME: LMOM2

TITLE: Compute the second moment of a chain about a line.

TYPE: Subroutine.

CALL FORM: CALL LMOM2(IDENT,X1,Y1,X2,Y2,SMNT)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
X1	I	R	-	X coordinate of first point
Y1	I	R	-	Y coordinate of first point
X2	I	R	-	X coordinate of second point
Y2	I	R	-	Y coordinate of second point
SMNT	O	R	-	Computed second moment

COMMON BLOCKS: XYCOMP,UCHAIN

SUBROUTINES CALLED: INITXY,UNPACK,MOM2A

DESCRIPTION: This subroutine computes second moment of a chain about a line defined by points (X1,Y1) and(X2,Y2).

LIMITATIONS: Chain IDENT must be a closed chain.

WORKING BUFFER USED: 1

NAME: LNAREA

TITLE: Compute the area between a chain and a line.

TYPE: Subroutine.

CALL FORM: CALL LNAREA(IDENT,X1,Y1,X2,Y2,AREA)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
X1	I	R	-	X coordinate of first point
Y1	I	R	-	Y coordinate of first point
X2	I	R	-	X coordinate of second point
Y2	I	R	-	Y coordinate of second point
AREA	O	R	-	Computed area

COMMON BLOCKS: XYCOMP,UCHAIN

SUBROUTINES CALLED: ECAREA,INITXY,UNPACK,MOM1A,CENTRD

LIMITATIONS: Chain IDENT must be a closed chain.

DESCRIPTION: This subroutine computes the area between a chain and a line defined by points (X1,Y1) and (X2,Y2).

WORKING BUFFER USED: 1

NAME: LNCHD

TITLE: Compute the max and min distance from a chain to a line.

TYPE: Subroutine.

CALL FORM: CALL LNCHD(IDENT,X1,Y1,X2,Y2,CLDMAX,CLDMIN,JMAX,JMIN)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
X1	I	R	-	X coordinate of first line point
Y1	I	R	-	Y coordinate of first line point
X2	I	R	-	X coordinate of second line point
Y2	I	R	-	Y coordinate of second line point
CLDMAX	O	R	-	Maximum distance between chain and line
CLDMIN	O	R	-	Minimum distance between chain and line
JMAX	O	I	-	Number of link at maximum distance
JMIN	O	I	-	Number of link at minimum distance

COMMON BLOCKS: none

SUBROUTINES CALLED: INITXY, ARRAY

DESCRIPTION: This subroutine finds the minimum and maximum distances between a chain and a line defined by points (X1,Y1) and(X2,Y2). JMAX and JMIN are numbers of directional links at the maximum and minimum distances, respectively.

WORKING BUFFER USED: 1,3.

NAME: MATCH

TITLE: Match two chain profiles.

TYPE: Subroutine.

CALL FORM: CALL MATCH(XX,YY,PROF1,N1,PROF2,N2)

PARAMETERS:

Name	I/O	Type	Dimension	Description
XX	O	I	-	Relative position in the x direction
YY	O	I	-	Relative position in the y direction
PROF1	I	I	2,N1	First profile sequence
PROF2	I	I	2,N2	Second profile sequence
N1	I	I	-	Dimension of PROF1
N2	I	I	-	Dimension of PROF2

COMMON BLOCKS: none

SUBROUTINES CALLED: none

DESCRIPTION: This subroutine computes the best match of two profile sequences PROFL1 and PROFL2. XX and YY are the number of grid positions to shift the first chain in the X and Y direction for the optimum match.

REFERENCES: Freeman, H., "On the template-layout problem", N.Y.U. Technical Report CRL-39, February 1975.

NAME: MAXMIN

TITLE: Compute the max and min extents of a chain in x and y directions.

TYPE: Subroutine.

CALL FORM: CALL MAXMIN(IDENT,XY)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain
XY	O	I	4	Array for the min/max values

COMMON BLOCKS: XYCOMP,UCHAIN

SUBROUTINES CALLED: UNPACK

DESCRIPTION: This subroutine computes the maximum and minimum coordinates of chain number IDENT in the x and y directions. The values are stored into the array XY(4) in the following order:

XY(1) - maximum x coordinate of chain
XY(2) - minimum x coordinate of chain
XY(3) - maximum y coordinate of chain
XY(4) - minimum y coordinate of chain

WORKING BUFFER USED: 1

NAME: MOM1

TITLE: Compute the first moment of a chain about 0,45,90,135 axes.

TYPE: Subroutine.

CALL FORM: CALL MOM1(IDENT,DEGREE,FMNT)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
DEGREE	I	I	-	Axis in degrees
FMNT	O	R	-	Computed first moment

COMMON BLOCKS: XYCOMP,UCHAIN

SUBROUTINES CALLED: INITXY,UNPACK

DESCRIPTION: This subroutine computes the first moment of a chain about 0, 45, 90, and 135 degree axes. The axis is selected by setting DEGREE to 0, 45, 90, or 135.

LIMITATIONS: The chain IDENT must be a closed chain.

WORKING BUFFER USED: 1

NAME: MOM2

TITLE: Compute the second moment of a chain about 0,45,90,135 axes.

TYPE: Subroutine.

CALL FORM: CALL MOM2(IDENT,DEGREE,SMNT)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
DEGREE	I	I	-	Selected axis in degrees
SMNT	O	R	-	Computed second moment

COMMON BLOCKS: XYCOMP, UCHAIN

SUBROUTINES CALLED: INITXY, UNPACK

DESCRIPTION: The second moment of a chain is found as the product of the area of the chain and the square of the distance from the axis. Thus the x and y components of each element with respect to the given axis is found. The moment of an area encircled in a clockwise sense above the x axis or in a counterclockwise sense below the axis is positive.

LIMITATIONS: Chain IDENT must be a closed chain.

WORKING BUFFER USED: 1

NAME: NPACK

TITLE: Pack a chain.

TYPE: Subroutine.

CALL FORM: CALL NPACK(IDENT,NB,START,STOP)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
NB	I	I	-	Number of working buffer
START	I	I	-	Index of the first location of the unpacked chain
STOP	I	I	-	Index of the last location of the unpacked chain

COMMON BLOCKS: CHAPMC,ICHAIN,PCHAIN,UCHAIN

SUBROUTINES CALLED: STLINK,GTLINK

DESCRIPTION: This subroutine creates a new packed chain from the contents of working buffer NB which contains a chain in the unpacked format. START is the index of the first used location in the working buffer NB. STOP is the last location used in the working buffer NB. IDENT is the identification number of the new chain .

NAME: OCTNUM

TITLE: Form an octal number from a sequence of links.

TYPE: Integer function.

CALL FORM: Variable=OCTNUM(BUFFER,I,N)

PARAMETERS:

Name	I/O	Type	Dimension	Description
BUFFER	I	I	M	One-dimensional array
I	I	I	-	Number of the first digit
N	I	I	-	Number of digits to form the number

COMMON BLOCKS: CHAPMC

SUBROUTINES CALLED: Implementation dependent. See appropriate appendix.

DESCRIPTION: This function returns the integer value (decimal) of a number formed from N octal digits stored in one dimensional array BUFFER as described for programs GTLINK and STLINK. The I-th octal digit is in the most significant position and the (I+N-1)-th octal digit is in the least significant position of the number.

LIMITATIONS: In order to stay within the array BUFFER:

$1 \leq I \leq NLW*M$

$1 \leq N \leq NLW*M-I+1$

NAME: OUTPUT

TITLE: Output chains to I/O unit 7 in format read by INPUT.

TYPE: Subroutine.

CALL FORM: CALL OUTPUT(IDENT)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number

COMMON BLOCKS: CHAPMC,PCHAIN,ICHAIN,IOBUFF

SUBROUTINES CALLED: GTLINK

DESCRIPTION: This subroutine writes a chain on a data file, which has been assigned FORTRAN logical I/O unit number 7. The chain data file will be output in the following format:

Columns	
1-5	Chain identification number
6	blank
7-72	Chain data
73-80	Sequence number

The sequence numbers begin at 100_{10} and are incremented by 100_{10}.

NAME: PDIST

TITLE: Compute the distance between two points on a chain.

TYPE: Subroutine.

CALL FORM: CALL PDIST(IDENT,P1,P2,DIST)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
P1	I	I	-	First chain vertex
P2	I	I	-	Second chain vertex
DIST	O	R	-	Distance between the two points

COMMON BLOCKS: XYCOMP,UCHAIN

SUBROUTINES CALLED: UNPACK

DESCRIPTION: This subroutine computes the straight line distance between two points on a chain defined by the vertices P1 and P2.

WORKING BUFFER USED: 1

NAME: PNTCHD

TITLE: Compute the max and min distances from a chain to a point.

TYPE: Subroutine.

CALL FORM: CALL PNTCHD(IDENT,XP,YP,DMAX,DMIN,LMAX,LMIN)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
XP	I	R	-	X coordinate of a point
YP	I	R	-	Y coordinate of a point
DMAX	O	R	-	Computed maximum distance
DMIN	O	R	-	Computed minimum distance
LMAX	O	I	-	Number of link at maximum distance
LMIN	O	I	-	Number of link at minimum distance

COMMON BLOCKS: XYCOMP,UCHAIN

SUBROUTINES CALLED: INITXY,UNPACK

DESCRIPTION: This subroutine computes the maximum and minimum distances between a chain and a point defined by (XP,YP). LMAX and LMIN are sequential numbers of chain directional links at maximum distance DMAX and at minimum distance DMIN, respectively.

WORKING BUFFER USED: 1

NAME: POLYGN

TITLE: Determine a polygonal approximation of a chain to a specified precision.

TYPE: Subroutine.

CALL FORM: CALL POLYGN (IDENT,XCOORD,YCOORD,ICL,IOP,TOL,IC)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
XCOORD	O	I	J	X coordinates of chain vertices
YCOORD	O	I	J	Y coordinates of chain vertices
ICL	O	I	J	Pointers to vertices of polygon
IOP	I	I	J	Stack used in approximation
TOL	I	R	-	Tolerance of the polygonal approximation
IC	O	I	-	Number of vertices of polygon

COMMON BLOCKS: XYCOMP,UCHAIN

SUBROUTINES CALLED: INITXY,UNPACK

DESCRIPTION: This subroutine computes the polygonal approximation of a a chain. XCOORD and YCOORD store the coordinates of the chain vertices. ICL stores pointers to the vertices of the polygon, i.e., XCOORD(ICL(J)) and YCOORD(ICL(J)) are the coordinates of the j-th vertex of the polygon, IC is the number of vertices of the polygon. TOL is set in a calling program to the required tolerance for the polygonal approximation.

LIMITATIONS: J is the number of vertex points in the chain.

WORKING BUFFER USED: 1

REFERENCES: Ramer, U., "An Iterative Procedure for the Polygonal Approximation of Plane Curves", Computer Graphics and Image Processing, 1, 1972, pp. 244-256.

NAME: PRINT

TITLE: Convert a chain to binary plane representation and print the binary plane.

TYPE: Subroutine.

CALL FORM: CALL PRINT(IDENT,IW)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
IW	I	I	4	Window to be printed

COMMON BLOCKS: CHAPMC,IOBUFF,PLANES

SUBROUTINES CALLED: Implementation-dependent. See appropriate appendix.

DESCRIPTION: This subroutine converts the chain IDENT into a binary plane and then outputs a window of the plane as specified by IW to FORTRAN logical unit number 6.

NAME: RESID

TITLE: Compute chain residue.

TYPE: Subroutine.

CALL FORM: CALL RESID(IDENT,LRES1,NLRES1,LRES2,NLRES2)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
LRES1	O	I	-	Number of first type links in residue
NLRES1	O	I	-	Number of links L1 in residue
LRES2	O	I	-	Number of second type links in residue
NLRES2	O	I	-	Number of links L2 in residue

COMMON BLOCKS: XYCOMP,UCHAIN

SUBROUTINES CALLED: UNPACK

DESCRIPTION: This subroutine computes the links in the residue of a chain. There are at most two types of links in the residue, which differ by unity, i.e. LRES1=LRES2+1(mod 8). Each type of residue link occurs 0 or greater number of times as specified by NLRES1 for type LRES1 and NLRES2 for type LRES2 of residue links.

WORKING BUFFER USED: 1

NAME: ROSCAL

TITLE: Rotate and scale a chain.

TYPE: Subroutine

CALL FORM: CALL ROSCAL (IDENT1, IDENT2, ANGLE, XSCALE, YSCALE)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT1	I	I	-	Identification number of the chain to be rotated and scaled
IDENT2	I	I	-	Identification number of the resulting chain
ANGLE	I	R	-	Rotation angle in degrees
XSCALE	I	R	-	Scale factor in x direction
YSCALE	I	R	-	Scale factor in y direction

COMMON BLOCKS: UCHAIN, WCHAIN

SUBROUTINES CALLED: UNPACK, NPACK, INITXY, BUILD, LINK

WORKING BUFFER USED: 1,3

DESCRIPTION: This subroutine rotates and scales a chain. The resulting chain is stored with the specified identification number. If the angle is a multiple of 90^{o} and the scale factors are both 1.0, then a very efficient algorithm is used.

NAME: STLINK

TITLE: Store a link in a packed format buffer.

TYPE: Subroutine.

CALL FORM: CALL STLINK(LINK,BUFFER,I)

PARAMETERS:

Name	I/O	Type	Dimension	Description
LINK	I	I	-	Octal digit
BUFFER	I	I	M	One-dimensional array
I	I	I	-	Desired sequential position of LINK in BUFFER

COMMON BLOCKS: CHAPMC

SUBROUTINES CALLED: Implementation dependent. See appropriate appendix.

DESCRIPTION: This subroutine inserts the octal digit LINK into the one dimensional array BUFFER as the I-th sequential octal digit in packed format with NLW octal digits per word.

LIMITATIONS: In order to stay within the array BUFFER:

$$1 \leq I \leq NLW_*M$$

NAME: UNPACK

TITLE: Unpack a chain.

TYPE: Subroutine.

CALL FORM: CALL UNPACK(IDENT,NB,FIRST,ISTART,ISTOP,EOCH)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
NB	I	I	-	Number of working buffer
FIRST	I	L	-	First part of chain to be unpacked
ISTART	O	I	-	Index of the first location of the unpacked chain
ISTOP	O	I	-	Index of the last location of the unpacked chain
EOCH	O	L	-	End of chain flag

COMMON BLOCKS: CHAPMC,PCHAIN,UCHAIN,ICHAIN,WCHAIN

SUBROUTINES CALLED: GTLINK,OCTNUM

DESCRIPTION: This subroutine unpacks a portion of a packed chain into working buffer NB. ISTART is the index of the first used location in the working buffer NB, ISTOP is the index of the last used location in the working buffer NB. The logical flag EOCH is.TRUE.if the end of chain indicator was reached.

The subroutine is called multiple times to unpack a long chain. FIRST is to be set to .TRUE. in the call if the first part of a chain is to be unpacked. It is to be set to .FALSE. in the call to unpack additional portions of the chain until EOCH is .TRUE.

NAME: VERTEX

TITLE: Convert a chain into lists of vertex coordinates.

TYPE: Subroutine.

CALL FORM: CALL VERTEX(IDENT,XCOORD,YCOORD,N,L)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
XCOORD	O	I	N	An array of x coordinates
YCOORD	O	I	N	An array of y coordinates
N	I	I	-	Dimension of XCOORD and YCOORD
L	O	I	-	Number of vertex coordinates in XCOORD and YCOORD

COMMON BLOCKS: XYCOMP,UCHAIN

SUBROUTINES CALLED: INITXY,UNPACK

DESCRIPTION: This subroutine converts a chain into two arrays. Array XCOORD contains the x coordinates of the vertex points of the chain and array YCOORD contains the y coordinates. Note that XCOORD(1) and YCOORD(1) contain the initial coordinates of the chain. If there are more than N-1 directional links in the chain the remaining links are skipped and a warning message is printed. L is set to the number of coordinate pairs in the XCOORD and YCOORD arrays.

WORKING BUFFER USED: 1

NAME: VISIBL

TITLE: Append visible-chain-follows signal code.

TYPE: Subroutine.

CALL FORM: CALL VISIBL(IDENT)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number

COMMON BLOCKS: STATUS,UCHAIN

SUBROUTINES CALLED: none

DESCRIPTION: The visible-chain-follows signal code '0402' is appended to chain IDENT which is currently being built by CHAP, by placing the number 12_{10} in the next location of the working buffer.

NAME: WHEX

TITLE: Compute chain width, height, 45 and 135 degree extents.

TYPE: Subroutine.

CALL FORM: CALL WHEX(IDENT,ITYPE,W)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
ITYPE	I	I	-	Type of computation desired
W	O	R	-	Computed value

COMMON BLOCKS: UCHAIN

SUBROUTINES CALLED: UNPACK

DESCRIPTION: This subroutine computes the width, height, 45 degree extent or 135 degree extent of a chain by specifying ITYPE to be

1 - width
2 - height
3 - 45 degree extent
4 - 135 degree extent

WORKING BUFFER USED: 1

NAME: XCOORD

TITLE: Append an x coordinate to a chain.

TYPE: Subroutine.

CALL FORM: CALL XCOORD(IDENT,X)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
X	I	I	-	X coordinate

COMMON BLOCKS: STATUS,UCHAIN

SUBROUTINES CALLED: none

DESCRIPTION: The x coordinate value and signal code is appended to a chain with identification number IDENT which is built by CHAP.

LIMITATIONS: $0 \leq X \leq 32768_{10}$

NAME: XPROFL

TITLE: Compute a chain profile along x axis.

TYPE: Subroutine.

CALL FORM: CALL XPROFL(IDENT,FLAG,PROFL,N,LIST,NN)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
FLAG		L	-	Flag indicating upper profile (.TRUE.) or lower profile (.FALSE.) desired
PROFL	O	I	2,N	Array of profile pairs
N	I	I	-	Dimension of array PROFL
LIST	T	I	NN	Array to store chain
NN	I	I	-	Dimension of array LIST

COMMON BLOCKS: XYCOMP

SUBROUTINES CALLED: ARRAY,MAXMIN

DESCRIPTION: This subroutine forms profile pairs of a chain-coded template. The profile pairs are computed relative to the enclosing rectangle made of x and y maximum and minimum positions of the chain. When FLAG is true the profile is computed along the upper edge, when false the profile is computed along the lower edge of the template.

REFERENCES: Freeman, H., "On the template-layout problem", N.Y.U.
Technical Report CRL-39, February 1975

WORKING BUFFER USED: 1,3

NAME: YCOORD

TITLE: Append a y coordinate to a chain.

TYPE: Subroutine.

CALL FORM: CALL YCOORD(IDENT,Y)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
Y	I	I	-	Y coordinate

COMMON BLOCKS: STATUS,UCHAIN

SUBROUTINES CALLED: none

DESCRIPTION: The y coordinate and signal code is appended to a chain with identification number IDENT which is currently being built by CHAP.

LIMITATIONS: $0 \leq Y \leq 32768_{10}$

8. AN EXAMPLE OF A CHAP USER PROGRAM

A simple CHAP user program is listed on the following page. The program reads an input chain data file which consists of 2 records. The first contains a '1' in column 5 and, starting in column 7, the chain "12123000765455404400". The second record consists of a zero in column 5. The expected output is shown in the program listing. The chain is listed, then the image of the chain is printed from a 41x6 point window of the CHAP binary array as specified by window size IW. Then the enclosed area, length, first moment about the y axis and second moment about the x axis of the chain are computed and printed. The output of the example program is given on page 8.3.

```
>C
>C
>C  AN EXAMPLE OF CHAP PROGRAMMING
>C
>C
>      IMPLICIT INTEGER(A-Z)
>      REAL L,S,FMNT,SMNT
>      DIMENSION IW(4)
>      DATA IW / 0,40,0,5 / , LCF / 0 /
>C
>C  READ IN ONE SAMPLE CHAIN, WITH IDENT. NO. = 1
>      CALL INPUT(0)
>      ID = 1
>C
>C  LIST THE CHAIN
>      CALL CHLIST
>C
>C  PRINT THE CHAIN
>      CALL PRINT(ID,IW)
>C
>C  FIND THE ENCLOSED AREA
>      CALL ECAREA(ID,S)
>      WRITE(6,100) ID,S
> 100 FORMAT(31H-THE ENCLOSED AREA OF CHAIN NO.,I3,4H IS ,F7.2/)
>C
>C  FIND THE LENGTH OF THE CHAIN
>      CALL LENGTH(ID,L,LCF)
>      WRITE(6,110) ID,L
> 110 FORMAT(24H THE LENGTH OF CHAIN NO., I3, 4H IS , F7.2/)
>C
>C  FIND THE FIRST MOMENT ABOUT THE Y AXIS
>      CALL MOM1(ID,90,FMNT)
>      WRITE(6,120) FMNT
> 120 FORMAT(31H THE FIRST MOMENT ABOUT Y AXIS:, F7.2/)
>C
>C  FIND THE SECOND MOMENT ABOUT THE X AXIS
>      CALL MOM2(ID,0,SMNT)
>      WRITE(6,130) SMNT
> 130 FORMAT(32H THE SECOND MOMENT ABOUT X AXIS:, F7.2/)
>C
>C
>C  EXPECTED OUTPUT:
>C  CHAIN LISTING-       12123 00076 54554 04 00
>C  ENCLOSED AREA-       11.50
>C  LENGTH-              17.90
>C  FIRST MOMENT (Y AXIS)   -32.17
>C  SECOND MOMENT-          121.08
>C
>      STOP
>      END
```

```
#EXECUTION BEGINS



   CONTENTS OF THE CHAIN BUFFER


 CHAIN ID NUMBER =      1

 12123  00076  54554  0400

 END OF BUFFER





  CHAIN FILE NO.      1

   XXXX
    X  X
    X  X
   X XX
   XX
  XX

 THE ENCLOSED AREA OF CHAIN NO.  1 IS    11.50

 THE LENGTH OF CHAIN NO.  1 IS    17.90

 THE FIRST MOMENT ABOUT Y AXIS: -32.17

 THE SECOND MOMENT ABOUT X AXIS: 121.08
#EXECUTION TERMINATED
#
```

9. REFERENCES

1. Freeman, H., "On the encoding of arbitrary geometric configurations," IRE Trans. on Elect. Computers, EC-10, (2), June 1961, 260-268.

2. ___________, "Techniques for the digital computer analysis of chain-encoded arbitrary plane curves," Proc. Nat'l. Elect. Conf., 17, 321-421, October 1961.

3. ___________, "Boundary encoding and processing," in Picture Processing and Psychopictorics, ed. by B. Lipkin and A. Rosenfeld, Academic Press, Inc., New York, 1970.

4. ___________, "Computer processing of line-drawing images," Computing Surveys, 6, (1), March 1974, 57-97.

5. Johnston, E.G., "The PAX User's Manual," Computer Science Center, University of Maryland, College Park, MD 20742, June 1972, (NTIS AD745 973).

10. APPENDICES

A. Multiple-Computer Version

B. IBM 360 Version

Appendix A. MULTIPLE-COMPUTER VERSION

Most of the CHAP subroutines are totally independent of the computer on which CHAP is to be run. All that is required is a standard ANSI FORTRAN IV compiler. The only exceptions are the chain packing and unpacking routines, which depend on the word length of the computer, as well as some minor variations in the particular FORTRAN versions. It is planned to maintain a single master version of CHAP. This version will consist of the union of all the statements required for all the different computer systems on which CHAP is to be implemented. It will not be an executable version. Special control cards, identified by the letter "M" in column 1 will be used to indicate the computer systems to which the statements that follow apply. For each computer system, a unique column, beginning with column 10, is assigned in the M-cards. An entry (any alphanumeric character other than 'blank') in one or more columns indicates that all the cards that follow (until the next M-card is encountered) apply to all the computer systems that have an entry in their assigned column. (The entries may be coded to facilitate readability).

Statements that apply to <u>all</u> computer system versions are preceded by an M-card that either contains an entry in <u>every</u> assigned computer column beginning with column 10, or contains an A in column 9. In the latter case, any entries in columns 10 and above are ignored.

To generate a CHAP version for a particular computer system, the master version is first pre-processed by a small program to (1) delete all statements for which the preceding M-card either does not have an A in column 9 or does not have an entry in the column corresponding to the desired computer version, and (2) delete all M-cards. The result will be an executable FORTRAN IV program that will run on the desired computer system.

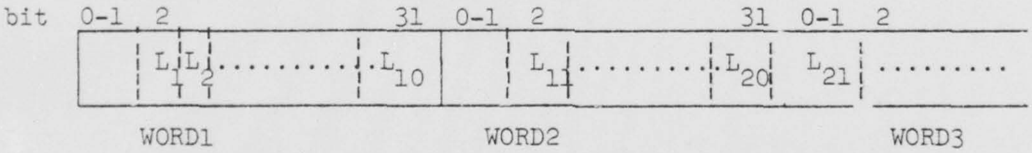
The master version will be centrally maintained, revised, and extended as found necessary or desirable.

Appendix B. IBM 360/370 VERSION

This version of CHAP was written for an IBM 360 computer and compiled by a FORTRAN (G) compiler. It was tested on the IBM-360/67 at RPI, using the Michigan Terminal System.

Three IBM assembly language integer functions are used. They are IAND, IOR, and SHIFT, and are described on the following pages. They are used in the process of packing and unpacking chains. The memory organization of CHAP packed chains is shown in the figure below. Ten (10) links are packed in one 32-bit word in this version. Bits 0 and 1 of each word are not used.

BITS/WORD (NBW)	LINKS/WORD (NLW)
32	10



The Block Data program CHAPBD must be link-edited with the user's main program to allocate the required memory space correctly. The working buffers (unpacked) are allocated 600 words in total, and the packed buffer is allocated 500 words in this version. The IBM version of CHAPBD is listed in this appendix.

NAME: YPROFL

TITLE: Compute chain profile along y axis.

TYPE: Entry to subroutine XPROFL.

CALL FORM: CALL YPROFL(IDENT,FLAG,PROFL,N,LIST,NN)

PARAMETERS:

Name	I/O	Type	Dimension	Description
IDENT	I	I	-	Chain identification number
FLAG	I	L	-	Flag indicating right (.TRUE.) or left profile (.FALSE.) is desired
PROFL	O	I	2,N	Array for profile pairs
N	I	I	-	Dimension of array PROFL
LIST	TO	I	NN	Array to store chain
NN	I	I	-	Dimension of array LIST

COMMON BLOCKS: XYCOMP

SUBROUTINES CALLED: ARRAY,MAXMIN

DESCRIPTION: This subroutine forms profile pairs of a chain-coded template. The profile pairs are computed relative to the enclosing rectangle made of x and y maximum and minimum positions of the chain. When FLAG is true the profile is computed along the right edge, when false the profile is computed along the left edge of the template.

REFERENCES: Freeman, H., "On the template-layout problem", N.Y.U. Technical Report CRL-39, February 1975

WORKING BUFFER USED: 1,3

ADDITIONAL LIST OF GLOBAL NAMES

The following is a list of the global names which are defined in the IBM-360/370 version of CHAP. They are all integer functions (IF) and are described on pages B.3-B.5.

Name	Type	Description
IAND	IF	The logical and of two variables.
IOR	IF	The logical or of two variables.
SHIFT	IF	Left circular shift of an integer variable.

The following list is of the CHAP routines which call the implementation dependent routines and the routines they call.

CHAP routine	Routines called
APLANE	SHIFT, IOR
BPLANE	SHIFT, IOR
BPRINT	SHIFT, IAND
CHPLOT	----------
GTLINK	SHIFT, IAND
OCTNUM	SHIFT, IAND
PRINT	SHIFT, IAND
STLINK	SHIFT, IAND, IOR

NAME: IAND

TITLE: Form bit-by-bit logical AND of two variables.

TYPE: Integer function.

CALL FORM: Variable = IAND(WORD1, WORD2)

PARAMETERS:

Name	I/O	Type	Dimension	Description
WORD1	I	I	-	An integer variable
WORD2	I	I	-	An integer variable

COMMON BLOCKS: none

SUBROUTINES CALLED: none

DESCRIPTION: IBM assembly language routine for the IBM 360 implementation of CHAP. Assembled by ASSEMBLER (F). The logical AND of the two parameters is returned.

NAME: IOR

TITLE: Form bit-by-bit logical OR of two variables.

TYPE: Integer function.

CALL FORM: Variable = IOR(WORD1, WORD2)

PARAMETERS:

Name	I/O	Type	Dimension	Description
WORD1	I	I	-	An integer variable
WORD2	I	I	-	An integer variable

COMMON BLOCKS: none

SUBROUTINES CALLED: none

DESCRIPTION: IBM assembly language routine for the IBM 360 implementation of CHAP. Assembled by ASSEMBLER (F). The logical OR of the two parameters is returned.

NAME: SHIFT

TITLE: Left circular shift of an integer variable.

TYPE: Integer function.

CALL FORM: Variable = SHIFT (WORD,N)

PARAMETERS:

Name	I/O	Type	Dimension	Description
WORD	I	I	-	An integer variable
N	I	I	-	Number of bits to shift $0 \leq N \leq 30$

COMMON BLOCKS: none

SUBROUTINES CALLED: none

DESCRIPTION: Left circular shift of the 30 lower bits of an integer variable is performed. The two leading bits of the result are set to zero.

IBM assembly language routine for the IBM 360 implementation of CHAP. Assembled by ASSEMBLER (F).

6/77

```
C
C  ***********************
C      BLOCK DATA CHAPBD
C  ***********************
C
      BLOCK DATA
C
C              CHAP COMMON BLOCK ALLOCATION AND INITIALIZATION
C
C              THIS BLOCK DATA PROGRAM ALLOCATES AND INITIALIZES
C              ALL COMMON BLOCKS OF CHAP
C
C              THIS BLOCK DATA PROGRAM MUST BE COMPILED OR
C              LINK-EDITED TOGETHER WITH THE MAIN PROGRAM
C
      IMPLICIT INTEGER(A-Z)
C
C
      COMMON /CHAPMC/ NBW,NLW,EX,BLANK,DIGITS
      COMMON /XYCOMP/ AX,AY
      COMMON /PCHAIN/ BUFFER
      COMMON /UCHAIN/ UNBUFF
      COMMON /ICHAIN/ UCH,MAXNCH,NCH,CHDIR
      COMMON /WCHAIN/ WBASE
      COMMON /PLANES/ NX,NY,PLANE
      COMMON /IOBUFF/ IO
      COMMON /STATUS/ ID,SX,SY,SSTART,SSTOP,LLINK,PART,SEOCH
C
C              CHARACTER CODES OF THE OCTAL DIGITS 0 TO 7
      DIMENSION DIGITS(8)
C              X AND Y COMPONENTS OF CHAIN LINKS
      DIMENSION AX(8),AY(8)
C              CHAIN BUFFERS
      DIMENSION BUFFER(500)
      DIMENSION CHDIR(60)
C              WORKING BUFFERS
      DIMENSION UNBUFF(600)
C              BINARY PLANE
      DIMENSION PLANE(480)
C
C              INPUT/OUTPUT BUFFER
      DIMENSION IO(131)
      DIMENSION UCH(9)
      DIMENSION WBASE(4)

C
C              STATUS ARRAYS
      LOGICAL SEOCH(3)
      DIMENSION ID(3),SX(3),SY(3),SSTART(3),SSTOP(3),LLINK(3),PART(3)
C
C              MACHINE DEPENDENT CONSTANTS
C              NBW                 NUMBER OF BITS PER WORD
C              NLW                 NUMBER OF LINKS PER WORD
      DATA NBW,NLW /30,10/
C NOTE THAT IN THE IBM VERSION OF CHAP BITS 2-31
C ARE USED TO STORE THE LINKS OR BINARY PLANE
C
C              CHARACTER SYMBOLS
      DATA  EX,BLANK,DIGITS / 'X',' ','0','1','2','3','4','5','6','7' /
C              BASE INDEXES OF THE THREE WORKING BUFFERS
      DATA WBASE / 1,201,401,601 /
C              X AND Y COORDINATES OF LINKS
      DATA AX / 1,1,0,-1,-1,-1,0,1 /, AY / 0,1,1,1,0,-1,-1,-1 /
C              PICTURE PLANE
      DATA NX,NY /120,120/
C
C              CHAIN FILE CATALOG
      DATA UCH,MAXNCH,NCH,CHDIR /9*0,20,0,60*0/
C
C      -       STATUS VALUES
       DATA ID,SX,SY,SSTART,SSTOP,LLINK,PART /21*0/
       DATA SEOCH /3*.TRUE./
      END
```